IN THE SPECIFICATION:

On page 2, please amend the paragraph beginning on line 5, and the paragraph following thereafter, as follows:

--A learning system according to the invention ("Azalea"), to which the concept of template automaton is introduced, collects many [["]]expected examples of a variety of learners, including "right" and "wrong" responses. As an efficient error diagnosis engine in the language learning system, the NLP technology of an HCS (heaviest common sequence) or an LCS (longest common sequence) algorithm plays a decisive role. Those examples embedded into the template ~~is~~ are used for the diagnosis and analysis of learners' responses. The diagnosis is to be implemented by selecting a closest path from among a huge number of candidate paths in template databases to the learner's input sentences. The authoring task of building a template corpus consisting of well-formed model translations and ill-formed erroneous sentences is quite labor-intensive, taking up considerable time.

The new system of the invention not only simplifies or reduces the authoring task of template generation, which otherwise is time-consuming (refer to, JP-A No. 2002-49617 by Naoyuki Tokuda, Liang Chen, Hiroyuki Sasai, et al), but also is effective for the improvement of system performance. The first reason why the

2

introduced template-template architecture can simplify the system and improve the performance is that the architecture makes it possible to integrate many different templates into a single template-template and vice versa, that is to extract many different templates therefrom by applying extracting rules assigned to some of the transition nodes of a single template. The second reason is that the introduced ~~buggy~~ error rules have the function of automatically distinguishing and classifying erroneous learners' responses and, accordingly, generating ~~bugs~~ errors therefrom. The importance of the NLP (natural language processing) techniques in the development of the system is obvious because a parser is used for examining the learner's structure free-format response and the semantic structure is examined by checking the learner's response in a character string against the semantically equivalent path of the provided template data base.--

On page 3, please amend the last paragraph as follows:

--The template-template structure based on the new extracting rules or ~~buggy~~ error rules is expected to play an important role in many applications, when used in any system selected from learning systems having character input and interaction means, voice-based call centers or voice-enabling

3

portal systems, and any systems focusing on more enhanced human computer interfaces implementing more natural human-computer interactions between the system and humans.--

On page 4, please amend the paragraph beginning on line 10, and the paragraph following thereafter, as follows:

--2. ~~Exploiting~~ By exploiting the expanding power of the ~~buggy~~ error rules introduced, many erroneous expressions and/or many erroneous syntactic structures can now be described automatically with consistency so that the new template-template can be expanded ~~to a "thick"~~ into a fuller, more comprehensive template-template form, because of its automatic capability of describing the error ~~bug~~ taxonomy. The language teacher need not be concerned with the details of classifying erroneous translations at the time of authoring the template-template, thus bearing a reduced burden of error taxonomy, when evaluating and clustering the characteristics of the particular errors of learners.

3. The HCS matching algorithm can be developed so that the algorithm matches the input sentence against the simpler template-template directly, thus reducing the spatial and temporal calculation load in matching processes ~~in finding~~ to find the best matched paths from among all the possible ~~path~~ paths of all the

extracted templates without actually expanding the template-template.--

On page 5, please amend the paragraph beginning on line 6, and the paragraph following thereafter, as follows:

--First the term "template-template" will be defined below. The template-template is defined as a special template where some of the nodes are marked with extracting rule-associated symbols allowing the template-template to be expanded into many templates, or a so-called ~~large~~ "larger" template if a set of non-connected templates are regarded as one template. Such a set of disconnected templates ~~can now allow~~ allows a variety of possible translations of a single L1 sentence to form a single large ~~single~~ template-template comprising a group of translated L2 sentences. Being an extended template, the template-template scheme allows one or more templates to be extracted from the template-template ~~to extract one or more of the templates~~.

Typically, an extracting rule is always associated with a set of symbols, say $\{s_1, s_2, \ldots, s_n\}$, and each of the symbols is assigned one or more nodes in the templates. These associated symbols are assigned with one or more values whose function is to represent the style of the nodes that will appear in a template or

templates extracted from the template-template. These symbols are herein referred to as "label symbols." The symbols related to a single rule are called "related symbols." Related symbols should have certain restrictions. As a typical restriction, for a given $s_{i=1}$, s_k must often be restricted to 2, or to some positive ~~integers~~ <u>integer</u> other than 1. If the value of s_i depends on the values assigned to a set of the other symbols, the choice of <u>the</u> value of s_i is called a required choice of the other symbols.--

On page 7, please amend the paragraph beginning on line 4, as follows:

--As in the Type A Rule, the <u>Type B</u> ~~template-template~~ rule imposes the condition that the nodes marked with $PPRP_i$ and the other nodes marked with $PPRP_i$ (i being any integer) appearing in a set of templates must respectively take on the form of the personal pronoun and the personal pronoun possessive form of the pronouns, as required by the natural language grammar of the pronouns. Given $PPRP_i$ (or PPR_i), the required values of PPR_i (or $PPRP_i$) must be defined by the natural language grammar of pronouns.--

On page 7, please amend the last paragraph as follows:

--A ~~type~~ <u>Type</u> D Rule imposes the condition that among all the nodes of the template-template marked by $CHOi_1$, $CHOi_2$, . . . , $CHOi_k$,, one and only one set of nodes can appear in any of the templates extracted from the template-template. Here a different i represents a different Type D Rule. Hence, $CHOi_j=0$ implies that the nodes marked by $CHOi_j$ do not appear, while $CHOi_j=1$ implies that the specified nodes now appear. Obviously, if 1 is assigned to one $CHOi_k$, then 0 should be assigned to all the other $CHOi_j$.--

On page 8, please amend the heading on line 7, and the paragraphs following thereafter through page 8, line 6, as follows:

--The ~~Buggy~~ <u>Error</u> Rules for Expanding Template-Template

--A ~~buggy~~ <u>An error</u> rule here is defined as a production rule of<u>, or a rule for producing,</u> common syntactically erroneous expressions which are characterized by possible deviations from syntactically correct expressions.

To be specific, consider the following form of <u>an error</u> ~~a buggy~~ rule:

$$H_1H_2 \ \ldots \ H_N \rightarrow R_1R_2 \ \ldots \ R_M$$

where H_1H_2 . . . H_N represents a set of nodes tracking the syntactically correct path of any template-templates, or a set of

grammatical part-of-speech tags representing basic components or segments of a correct expression. R_1R_2 ... R_M is the set of nodes which represents a typical erroneous expression whose correct form is H_1H_2 . . . H_N. It is immediately seen that errors are identified by deviations from the correct paths of the template-template. Here is an example:

EX VBP $\rightarrow$ EX VBZ (Here EX represents existential such as "there is,"; VBP, verb for 1st and 2nd person present[[,]]; VBZ, verb, 3rd person singular present). This example implies that a syntactically correct expression "*there are* 5 books" is used erroneously by students who misunderstood the subject-verb alignment, resulting in an erroneous expression of "*there is* 5 books" in this example.--

On page 9, please amend the paragraph beginning on line 15, as follows:

--Figure 4 is a diagram showing a template 2 expanded for a sentence meaning "Japan is dotted with beautiful gardens nationwide".--

On page 10, please amend the last heading on the page and the paragraphs following thereafter through page 12, line 6, as follows:

--Example of Template-Template, Template-Template Expanded by ~~Buggy~~ Error Rules, and Templates Extracted from Template-Template

In the embodiment of the invention, a template-template for English translations of a Japanese sentence meaning "Japan is dotted with beautiful gardens nationwide." is constructed at first. The ~~numeral~~ numerals shown in Figure 1 and other figures represent the weights of the respective words ~~word~~ for enhancing the relative importance of the respective words in the sentence. The default weights of the words in the template are set to 1, and they must be assigned in accordance with the importance of the words as judged by experts in the field. Reference is made to JP-ANo. 2002-49617 by Naoyuki Tokuda, Liang Chen and Hiroyuki Sasai, for a detailed explanation. The symbols within "[" and "]" are the part-of-speech tags. The nodes shown in the left edge are starting nodes.

Now by simply applying the ~~buggy~~ error rules listed above, the template-template of Figure 1 can be expanded ~~expand it~~ into the template-template of Figure 2.

This shows that a language teacher need not be concerned with the details of classifying many common errors when he/she is constructing the template-template, since the <u>error</u> ~~buggy~~ rules can generate taxonomization of ~~bugs~~ <u>errors</u>, automatically allowing these erroneous expressions to be built into the template-template.

Now, by applying the Type A rule, it is easy to see that it is possible to extract a template as shown in Figure 3 from the template-template of Figure 2 by allowing the nodes marked with AP_1 to appear in the template and accordingly by deleting the nodes marked by NAP_1 of Figure 2~~, as well as~~<u>. Similarly, it is possible to extract</u> the template as shown in Figure 4, this time by deleting the nodes marked by AP_1 of Figure 2, and accordingly letting the nodes marked with NAP_1 <u>in Figure 2</u> appear in the template.--

On page 14, please amend the paragraph beginning on line 7, as follows:

--In the method of the present invention, a closest path is found from among all the valid paths of the templates that could be extracted from template-template with extracting rules (but without ~~buggy~~ <u>error</u> rules) directly, <u>and</u> without physically extracting all the templates from the template-template. It is necessary to expand the ~~buggy~~ <u>error</u> rules-embedded

template-template first so that the template-template does not
include any ~~buggy~~ <u>error</u> rules before such a matching takes place.
This can be performed in the steps of Figure 2 described above.--